SRS

spectra research systems

555 SPARKMAN DRIVE, SUITE 608, HUNTSVILLE, ALABAMA 35805 (205) 830-0375

(NASA-CR-161780) COAL CONVERSION SYSTEMS DESIGN AND PROCESS MODELING. VOLUME 2: INSTALLATION OF MPPM ON THE SIGNAL 9 COMPUTER Final Report (Spectra Research Systems, Inc.) 43 p HC A03/MF A01 CSCL 21D G3/28

N81-25234

Unclas
26536

SRS

spectra research systems

SOUTHEASTERN OPERATIONS
555 SPARKMAN DRIVE, SUITE 608, HUNTSVILLE, ALABAMA 35805
(205) 830-0375

SRS/SE-TR81-101

COAL CONVERSION SYSTEMS DESIGN AND PROCESS MODELING

FINAL REPORT

JUNE 1, 1981

VOLUME II

INSTALLATION OF MPPM ON THE SIGMA 9 COMPUTER

PREPARED FOR:

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
GEORGE C. MARSHALL SPACE FLIGHT CENTER
MARSHALL SPACE FLIGHT CENTER, ALABAMA 35812

UNDER CONTRACT NUMBER NAS8-34264

"LEGAL NOTICE"

"This report was prepared by the organization(s) named below as an account sponsored by the Tennessee Valley Authority. Neither TVA, the organization(s) named below, nor any person acting on their behalf: (a) makes any warranty express or implied, with respect to the accuracy, completeness, or usefulness of the information contained in this report, or that the use of any information, apparatus, method, or process disclosed in this report may not infringe privately owned rights; or (b) assumes any liabilities with respect to the use of, method, or process disclosed in this report."

(Organization(s) that prepared this report:)

SPECTRA RESEARCH SYSTEMS

<u>TABLE OF CONTENTS</u>

LIST OF FIGURES

LIST OF TABLES

1.0 FOREWORD

This final report is submitted to the National Aeronautics and Space Administration (NASA), George C. Marshall Space Flight Center (MSFC) by Spectra Research Systems, Southeastern Operations, 555 Sparkman Drive, Suite 608, Huntsville, Alabama 35805, as fulfillment of the final report requirement of the add-on task to Contract Number NAS8-34264 entitled "Coal Conversion Systems Design and Process Modeling." This report consists of two volumes, Volume I: Application of MPPM and ASPEN Computer Models, and Volume II: Installation of MPPM on the SIGMA 9 computer.

The work reported in these documents was performed for the MSFC Coal Gasification Task Team under the technical guidance of Mr. Cary H. Rutland, Manager, Mr. Emsley T. Deaton, Chief-Systems Engineering Office, and Mr. Robert L. Middleton, Contract Officer Representative. The major thrust of the study was to provide the basic framework for the development of a coal gasification systems design, systems sizing, and evaluation computer model which could provide process block configurations, material and energy balance between major process blocks, systems level sizing algorithms, and optional process block sequencing. Results of the study were inputs for MSFC's support to the Tennessee Valley Authority Coal Gasification Project. The study was accomplished by Spectra Research Systems (SRS) with the Mittelhauser Corporation (MC) acting as a subcontractor.

Mr. Rodney Bradford was the project manager and was supported by the SRS technical and management staff which included Dr. D. David Marshall, project leader, Mr. John D. Hyde, Mr. C. Wendell Mead, Mr. Edward E. Montgomery, and Mr. David E. Marty.

The MC technical staff included Mr. William H. Seward, project leader, Mr. M. Dale Dowden, Mr. Ronnie Johnson, and Mr. William G. Yeich.

The key administrative support staff for the effort was Ms. Sherry Clark and Mrs. Kathryn Henson.

2.0 INTRODUCTION

The purpose of this Appendix is to document the work performed by SRS to install the MPPM program on the NASA SIGMA 9 computer. This Appendix contains:

- an overview of the MPPM system;
- a description of the development of MPPM;
- SRS's procedure for procuring the MPPM program from International Research and Technology Corporation;
- a description of the relevant differences between the MPPM resident IBM 370 computer and the NASA SIGMA 9 computer;
- a discussion of the work performed to install the MPPM programs on the SIGMA 9 computer at MSFC;
- a list of problems encountered and solutions used to overcome these problems;
- a summary of remaining work in the installation effort; and
- recommendations and comments on the installation of MPPM on the SIGMA 9 or on an IBM computer.

This Appendix is also designed to serve as a compendium of information concerning transferring the MPPM programs from an IBM 370 to any other computer. The relevant computer hardware features incorporated in the MPPM program are described and their implications on the transportability of the MPPM source code are discussed.

3.0 MPPM OVERVIEW*

A Materials-Process-Product Model of coal-process technology has been developed for the U.S. Department of Energy by International Research & Technology Corporation (IR&T), which is an application of a methodology, previously developed under a grant from the National Science Foundation by IR&T, for systematically analyzing an array of competing manufacturing technologies from economic, environmental, and energy policy perspectives. The model consists of:

- A data base for coal-related materials and coal conversion processes.
- An algorithmic structure that facilitates systematic evaluations in response to exogenously specified variables such as tax policy, environmental limitations, and changes in process technology and costs.

The model has been developed as an interactive program, with maximum flexibility for inclusion of new process data, revision of old data, and specification of exogenous data related to policy options. It is operational on DOE's Energy Information Agency computer at Rockville, Maryland, an IBM-370 series machine.

3.1 PROGRAM OBJECTIVE

The objective of this program was to develop a tool, the Materials-Process-Product Model (MPPM), which could assist policy analysts in DOE (and systems analysts in the energy industry) in the systematic evaluation and comparison of alternatives for converting coal into energy and material products, e.g., substitute natural gas (SNG), base-load electric power, methanol, etc., under conditions of technological uncertainty. The emphasis in implementing the model has been on providing a means for rapid, repetitive, first-order evaluations of alternatives in response to projected and actual advances in technology, variations in economic variables, and changes in environmental restrictions. Computed outputs for complete process systems include material balances, capital requirements, environmental residuals, product prices, and energy efficiencies. The model data base is keyed to functional operations in coal conversion, e.g., gasification, purification, product synthesis, and is readily accessible by the analyst for experimentation with the physical, chemical and economic parameters that define the available alternatives. The model has been installed on the DOE computer at Rockville, MD as an interactive, conversational, remotely accessed set of programs and files.

---

*See references in Table 5-1.

3.2 MPPM BACKGROUND

The project has proceeded through three phases. Phase I, begun under an NSF grant in 1974 and extended by DOE in July 1975 and completed in February 1976, was an exploratory program in which the areas of technology to be modeled were bounded, the level of process technology detail appropriate to the MPPM was defined, and the feasibility of adapting the materials-process-product analytical method of coal-processing technology was established. In addition, the data base organization and algorithm structure for an MPPM tailored to the needs of coal-process technology were defined. Phase II, completed in November 1977, produced an operating model and a sizable data base. Phase III, which began in February 1978, was devoted to expanding the data base for the model in the areas of direct combustion, emission control and advanced gasification technologies. A review and update of the entire data base continued, and some changes were made in the model algorithms to enhance their generality and accuracy, particularly in the areas of plant utility modeling and adjustments of plant size requirements. The interactive programming structure was expanded to improve user/program communications, provide more options to the user in model execution, and to make the model more "transparent" to the user.

The MPPM is based on the materials-process-product methodology developed at IR&T for the analysis of manufacturing industries. The basis for the methodology is the characterization of industrial processes by material flows. Materials conservation, or in chemical engineering terms a materials balance, including all raw materials, all products, and all waste streams is imposed on individual processes and process systems. Energy inputs and outputs are related to materials throughputs and transformations. Similarly, investment costs and operating costs are related to the material and energy flows. Systems for conversion of raw materials to desired final products, including all secondary functions for the treatment or processing of waste and by-product streams, are evaluated by aggregation of all costs.

The methodology and the original version of the MPPM were developed and applied to a segment of the chemical industry. Since the original development project, the model has been greatly expanded, perfected, and programmed into the interactive, conversationally accessed form.

The methodological sequence employed in materials-process-product analysis, as related to coal conversion technology, is:

- Definition of the functional operations that enter into coal conversion processes, and modeling of alternative methods to accomplish these functions.
- Compilation of all the feasible conversion processes that can be assembled from combinations of these alternative methods for accomplishing the functional operations.
- Systematic, iterative evaluation of all feasible conversion processes under a variety of policy and economic scenarios, environmental constraints, and projected future events.

The second and third steps in this sequence are accomplished by the MPPM, i.e., by the analyst exercising the computer programs from his remote terminal in an interactive and conversational mode. The first step in the sequence is the assembly of the data base for the MPPM by the selection and modification of data from the "master" data file.

3.3 MPPM CAPABILITIES

The MPPM can provide the policy analyst with rapid, first-order evaluations of the effects that various policy decisions and institutional limitations could have on the viability of coal conversion processes in the market place. Similarly, it can provide the process systems analyst with first-order evaluations of the significance of technological developments, e.g., the effect that improvements at the subsystem level have on overall system functioning and costs. In either case, the interactive, conversational format, and ready access to the data base, facilitates the analysis of the sensitivity of the results to data uncertainty.

As an analytical tool, the MPPM is intended to satisfy the needs of an analyst working at the policy analysis or systems analysis level of detail. At this level, engineering design and costing models are too detailed and time consuming, while large scale energy/economic models are not sensitive to process-related trade-offs. In the case of coal process technology, many processes are in the earliest stages of development, and cost and performance data may, in some cases, be fairly crude estimates. The interests of DOE, Fossil Energy, the sponsor of MPPM development, are in the analysis of comparative economic and environmental impacts of alternative technologies; the ability to easily

perform sensitivity analyses is thus central to the usefulness of the MPPM. Construction of an MPPM attuned to the specific interests of the program sponsor has, of course, been the goal of the project. However, the model should also be useful to analysts in industry who require an analytical tool that deals with processes at the systems level.

The MPPM offers a new approach to the selection of complete flow diagrams for processes for converting coal to desired end products. The selection of process steps is possible on a logical basis and with much less dependence on intuition and encyclopedic knowledge of al the components that make up a process. The development of this type of approach is becoming more essential because of the proliferation of results from the DOE research programs, and the difficulty in selecting the process components that come closest to the optimum for a complete flow diagram.

Comparison of the MPPM approach with the analytical process development methods now in use illustrates its differences.

The usual chemical engineering approach to the development of capital and operating costs for a process such as coal conversion is to select, on a best judgment basis, a flow diagram, and calculate mass and energy balances from which sizes for all equipment can be estimated. From the equipment sizes the capital cost can be determined for each item. The sum of these items, coupled with proper allowances for foundations, piping, instruments, and auxiliaries, is then combined to estimate the total cost of the plant.

As processes become more complex this method of estimating is more time-consuming and laborious. Also, in this procedure, items often arise for which this relatively simple method of analysis is not applicable. For instance, it is not applicable to the development of costs for such items as an oxygen plant or coal pulverizing and sizing equipment. Therefore in practically all chemical engineering costing procedures it is customary to estimate the cost of the novel portion of the plant on the basis indicated, and to add to this the costs of standard modules such as the power plant, gas purification, shift conversion, and methanation and others.

The MPPM follows this process to its most logical conclusion, by developing a data base of models of functional operations, representing much of the present knowledge of conversion technology, then combining them to arrive at the feasible sequences available for a given overall conversion process. Each functional operation is so modeled that it can be modified within limits to to compatible with the other modules with which it is linked. If this is not possible, that module alternative is rejected from the flow diagram and other alternatives are selected.

Such a procedure is essential when dealing with large numbers of process alternatives, since cost calculations as now made are very costly for the complex, large-scale coal conversion processes. If a cost estimate is made and it is found desirable, for example, to change the gasification step, they may then necessitate important changes in many other sections of the plant, such as coal preparation, purification, shift conversion, etc. The result may be a completely new flow diagram requiring new mass and energy balances. This, in turn, necessitates a completely new cost calculation. Each attempt to better the process by changing the module alternatives within the process can result in extensive modifications and costly calculations.

It should also be noted that the MPPM will allow the investigation and testing of many alternatives in a given flow diagram even though the calculations are performed by investigators whose detailed knowledge may be limited with respect to some of the process steps. This means that models representing alternatives for a given functional operation may be defined by specialists in this specific area with respect to how and where each can be used, along with its limitations, and this functional module may then be used by investigators whose knowledge in this area is relatively limited.

The MPPM is not intended to replace present detailed costing methods on a specific plant design. Because of the expense of detailed cost estimating, including the process optimization and design work that must be done first, it is practical to undertake detailed cost estimates only when a flow diagram is frozen. The MPPM serves no such purpose, but is intended as an investigative procedure which would be particularly useful in the present formative state of development of energy conversion processes.

4.0 MPPM PROGRAM STRUCTURE

The structure of the MPPM as currently installed on the DOE IBM 370 computer located at the Rockville, Maryland facility is described by the diagram in Figure 4-1. The model is composed of a logic structure which can operate with any consistent data base. This is shown above the dotted line in the diagram. This part of the model is "process independent" in the sense that any process set that can be modularized and described in terms of algebraic input/output equations and cost equations can be reduced to a data base that the model logic set manipulates. This model logic will find and evaluate all possible sequences. of modularized processing steps that lead from defined inputs to defined outputs. The size of the data base in terms of alternative process descriptions or defined inputs and outputs is completely open-ended as far as the model logic is concerned. This model logic structure is completely operative.

The data base, or process dependent part of the model, as presently assembled encompasses thirty basic process functions which are required to convert coal into ten defined products. These functional modules are implemented by a total of forty basic alternatives (at least one per functional module) plus several variants of these and several no-cost "pass-through" variants. This data base is internally consistent and debugged. It implicitly contains more than three hundred viable complete processes (so-called economic unit processes or "EUP's") which are culled from a few thousand combinatorial possibilities.

This data base is structured for ease of expansion, contraction, or modification as required by a particular study. Additions or contractions can be at the functional module level or at the alternative configuration level. Individual users copy from master files to their own data subsets, which they can modify at will and submit to the model logic for computations.

The materials file contains data on over a hundred materials, including ten defined products and four defined types of coal. Materials may be added to this file or material descriptions may be modified. Alternatively, the model logic allows changes to be made while running the model, e.g., the sulfur content of coal can be changed at the start of a Linking Model run and a series of evaluations made with sulfur content as a parameter. (Each new raw material and/or product specification defines a new set of EUP's which are catalogued in numerical sequence by the model).

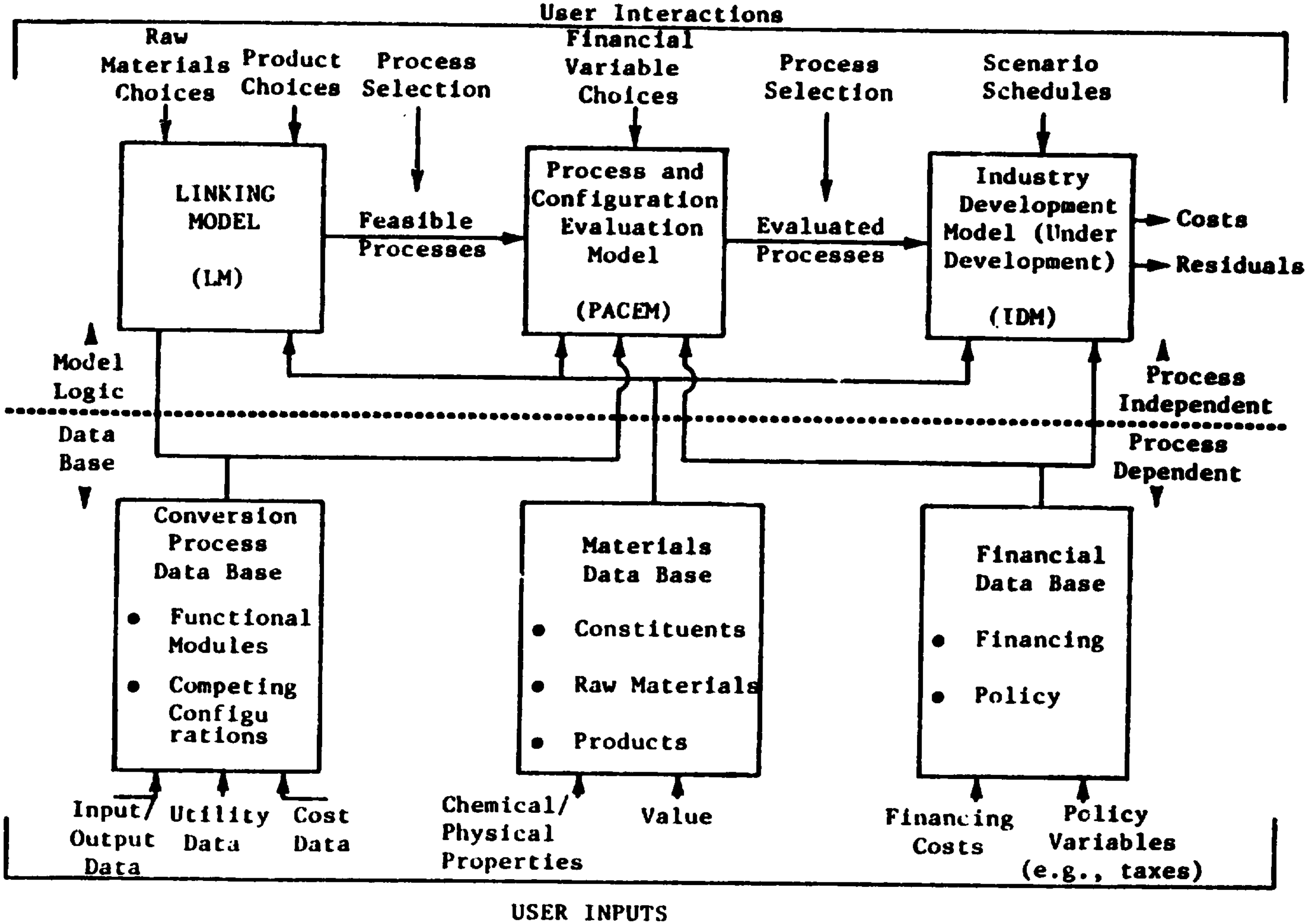

FIGURE 4-1 MPPM LOGIC STRUCTURE

The final output of the model includes summary reports which are printed at the remote terminal, and detailed formatted reports which are automatically routed to off-line high-speed printers. The former contains information such as the amount of coal used for auxiliary utilities generation, total investment capital required, and the "shadow price" for producing the chosen output material.* The latter contains all the information reported to the remote terminal plus detailed information such as the composition and quantity of materials passed from one configuration to the next, as well as costs and utility requirements of each configuration.

The model is thus useful for study of coal conversion processes at several levels of interest. Overall evaluations of processes using different economic assumptions can be handled, as well as systematic comparisons of the impact of technological advancements in a single functional module. Sufficient detail is reported to allow confirmation of technical feasibility and to expose interesting module interactions.

4.1 MPPM LOGIC CONCEPTS

The basic concept behand the MPPM is one of materials flow. Material is conserved in any industrial process; inputs are converted to useful outputs and waste. An "economic unit process" is defined as a transformation of a raw material into a useful, marketable product. Such a transformation can generally be divided into several steps or "functional modules". These functions are defined at a level of equipment aggregation which allows independent input/output characterizations of functional modules. In particular, "feedback" paths which give rise to input/output relationships with explicit coupling between configurations are avoided.

In general, a given, useful marketable product can be made by means of many feasible economic unit processes. These may differ because:

- they use different raw materials as starting points,
- they use different sequences of functional modules,
- they follow the same functional module sequences, but use different alternatives for one or more functional modules.

* Shadow price refers to the selling price required to meet all operating costs and yield a specific return on investment.

Different embodiments of a given functional module are called "competing configurations". Competing configurations are different methods for producing qualitatively the same principal output from the same principal input, although additional inputs needed, by-products produced, and economics may vary between configurations.

Given these definitions of economic unit processes, functional modules and competing configurations, it is clear that the latter serve as the basic building blocks. Any economic unit process consists of a sequence or chain of configurations. Whenever two economic unit processes differ (whether or not they use different raw material inputs, or different sequences of functional modules), they differ to the extent that at least one configuration in one economic unit process has been replaced by another.

It follows that all feasible economic unit processes for creating a useful, marketable product can be obtained by:

- finding all routes through the functional modules leading to the product,
- choosing from each functional module, in turn, compatible competing configurations until all possible combinations are exhausted.

These steps are accomplished by the Linking Model (LM) which assembles feasible unit economic processes from sequences of competing configurations; it is guided by a "Road Map" program, which defines allowable module connections, based on the road map files. They can be modified by the model user, to add newly defined functions to the model, as, for instance, might be required if a new end product is added. The "Road-Map" makes the search for feasible processes more efficient by eliminating unrealistic interconnections at the outset.

4.2 MPPM TERMINOLOGY

To further clarify the concepts, the MPPM terminology is illustrated in Figure 4-2. The upper diagram illustrates a chain of Economic Unit Processes involving substitute natural gas (SNG); the input and output of each block are marketable commodities (or at least would be if SNG were produced in commercial quantities)*. The middle diagram illustrates the breakdown of an

* Pipeline transportation is not currently modeled in the MPPM and is shown for illustrative purposes only. Transportation processes can be represented by joining the MPPM with other analytical models.

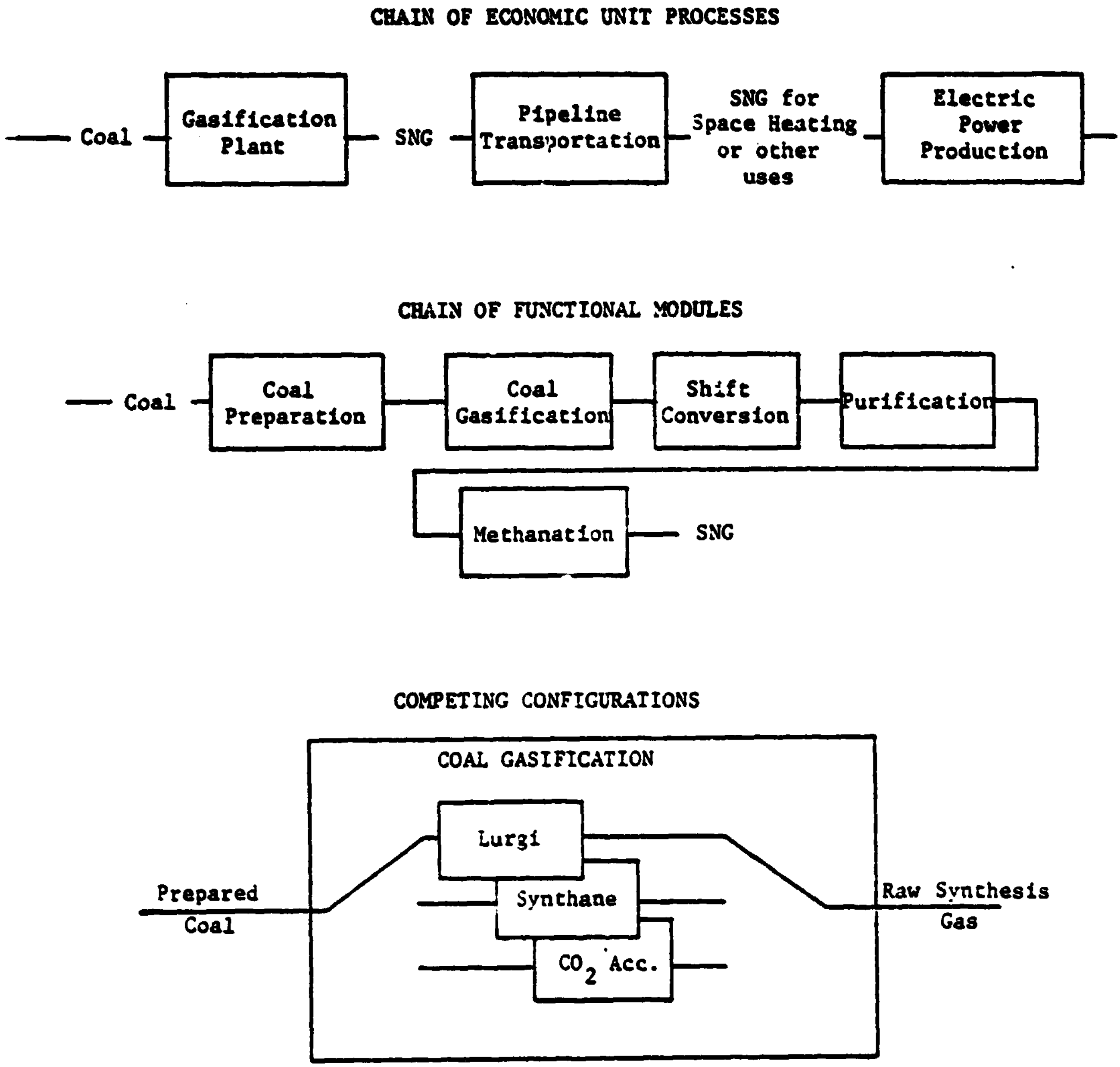

FIGURE 4-2 MPPM TERMINOLOGY

SNG production plant into functional modules. The bottom diagram shows the representation of coal gasification by a series of alternative methods, i.e., competing configurations. The MPPM assembles and evaluates gasification plants by selecting one competing configuration from each functional module, repeating the process until all feasible combinations have been evaluated.

"Feasibility" at the Linking Model stage is determined on a physical and chemical basis. Clearly, the number of combinations could become prohibitively large unless the concept of feasibility is inclusively and carefully applied.

For each competing configuration, a data base must be assembled. This consists of several items of a physical, chemical and economic nature:

- Input restrictions,
- Input/Output relations,
- Utility requirements,
- Capital costs and scale factors,
- Operating costs and scale factors.

In addition, there is a common data file on the physical and chemical properties and the value of all pertinent materials.*

Input materials to a configuration may be primary (e.g., a raw material like coal, or an important intermediate material like synthesis gas), or secondary (e.g., oxygen, process steam). They are coded accordingly. The distinctions are usually obvious but occasionally require engineering judgment. In addition, there may be utility inputs (steam, electric power, etc.). Outputs are also encoded as primary (useful intermediate or final products). In addition, there may be output to utilities in the form of recovered thermal energy. There is only one primary input and output for each configuration.

Input restrictions may put upper or lower bounds on the physical conditions (pressure, temperature) under which each configuration functions, on the range of constituents in a feed stream, or on the permissible amount of an inpurity in an input.

Input/output relations describe, as a function of physical and chemical variables, how the inputs are converted to outputs. They constitute a mathematical "model" of the transformation taking place in a given configuration.

---

* For materials which are not currently bought and sold at a market price on the open market, a price can be derived from the economics of their production. Such a price is often referred to as a "shadow price".

Capital and operating costs are needed to make an economic evaluation of each configuration, hence for each economic unit process as a whole.

The major characteristics of the Materials-Process-Product Model are summarized below:

- The model is based on materials flow and materials balance. For each configuration and for each economic unit process a materials balance can be performed to identify inputs, useful outputs and wastes. The same statement can be made concerning an energy balance.
- Each economic unit process is divided into a number of sequential functions with a single competing configuration accomplishing each function. Each configuration in turn, is represented in terms of input restrictions, input/output relations, and its estimated capital and operating costs.
- The modular structure of the MPPM, in terms of files of competing configurations, and general data, gives it a high degree of flexibility, and permits ready insertion of additional configurations, or of changes in configuration data.
- The model does not optimize manufacturing processes; configuration descriptions in the data base are optimized within the constraints imposed by the modular structure. Comparative evaluation of processes (EUPs) comprised of different competing configurations implies an optimization at the system level. However, specific criteria for optimization must be applied by the user of the model to the model outputs.

The structural elements of the MPPM are:

- A data base consisting of physical, chemical, and economic data for each configuration. From the physical and chemical descriptions, specialized listings are prepared that identify each configuration by primary input and output, and each input or output by its role. Also included are data files on the physical and chemical properties of materials and their value or cost data.
- A Linking Model (LM) which assembled feasible (on a physical and chemical basis) economic unit processes from sequences of competing configurations. The Linking Model contains within it, and is guided by, a "Road-Map" program, which establishes linkages or "Functional-Module Routes".
- A Process and Configuration Evaluation Model (PACEM) which computes value added and "shadow" prices (based on cost of materials and production) for the outputs of economic unit processes. This model allows economic variables, such as interest rates, taxes, subsidies, etc., to be varied exogenously. Effluent discharges and pollution atatement costs are also evaluated.

5.0 SOURCE CODE PROCUREMENT

MPPM was used to provide the basic structure and high-level modeling blocks for the systems level Materials and Energy Balance Simulation (MEBS) which could be used to evaluate alternative system designs. SRS, through a purchase order to IR&T, obtained the MPPM source code and related program documentation. A list of the documentation received is given in Table 5-1.

The documentation provided as the project Phase III output contains an executive summary, a description of the MPPM models and data base, an advanced gasification processes study, and a program users manual.

An IBM 370 generated 1600 bits per inch, non-labeled, 9 track magnetic tape was provided by IR&T based on SRS specifications. This tape contained the MPPM source code, the required MPPM data files, and the IBM 370 command structures which were used to combine the various program modules to produce a usable system. The tape consisted of 85 data sets (files). The first data set was the Job Control Language (JCL) stream used to create the tape. This first data set defined the name and location of each MPPM module as it appeared on the tape. These data set names and the name assigned during the SIGMA 9 installation are shown in Table 5-2. This tape and the MPPM documentation provided the basis for the installation of MPPM on the NASA SIGMA 9 computer. The manner in which the various data sets are used is described in Section 7.0, Installation Procedure on SIGMA 9.

TABLE 5-1

MPPM DOCUMENTATION RECEIVED

Phase III, Final Report, Volume I, Summary
Phase III, Final Report, Volume II, Appendices
Final Report for Project Phase III, Volume III, Users Guide to the MPPM
Phase III, Task 1 Report, Direct Combustion and Sulfur Emissions, Control and Recovery Processes, Volume I, Results
Phase III, Task 1 Report, Volume II, Part 1 - Process Data
Phase III, Task 1 Report, Volume II, Part 2 - Process Data
Phase III, Task 2 Report, Advanced Gasification Processes, Volume 1, Results
Phase III, Task 2 Report, Advanced Gasification Processes, Volume 2, Process Data

TABLE 5-2

MPPM DATA SETS

IBM 370 DATA SET NAME	SIGMA FILE NAME	IBM 370 DATA SET NAME	SIGMA FILE NAME
HE3.JCL.COPY.CNTL	F1	LINK.SIZER.CNTL	F43
MPPM.FUNMOD.RMAP.DATA	F2	LINK.UTIL1.CNTL	F44
MPPM.MATNODE.RMAP.DATA	F3	LINK.UTIL2.CNTL	F45
MPPM.ALTMAP.RMAP.DATA	F4	SELECT.REPORT.CNTL	F46
MPPM.ASPTAB.DATA	F5	WATERV2.FORT	F47
MPPM.NEW.MASTER.DATA	F6	DLNK31V1.FORT	F48
MAKMAT.MIX.DATA	F7	UTIL3MV3.FORT	F49
MAKMAT.SUB.DATA	F8	DSIZ29V1.FORT	F50
CONFIG.NAMES.CLIST	F9	UTIL2MV2.FORT	F51
COSTER.CLIST	F10	CONFIGV1.FORT	F52
COSTER.EQN.CLIST	F11	SELRETV1.FORT	F53
COSTER.MATL.CLIST	F12	COSTCOV3.FORT	F54
COSTER.REPORT.CLIST	F13	DMPEUPV1.FORT	F55
COSTER.REPORT.TERMINAL	F14	EQDOV1.FORT	F56
DELETE.FILES.CLIST	F15	EQDOXV1.FORT	F57
DMPEUP.CLIST	F16	EQDOZV1.FORT	F58
LINKER.CLIST	F17	JQUERYV1.FORT	F59
DUMP.FILES.CLIST	F18	UANDSDV3.FORT	F60
MAKFIG.CLIST	F19	MUTILV2.FORT	F61
MAKMAT.CLIST	F20	LINKV1.FORT	F62
MATEDT.CLIST	F21	LINK2V1.FORT	F63
UTIL3.EQN.CLIST	F22	LQUERYV1.FORT	F64
MERGER.CLIST	F23	SIZELV1.FORT	F65
MISSING.ASPECTS.CLIST	F24	MAKMATV1.FORT	F66
MISSING.LINKS.CLIST	F25	MATEDTV1.FORT	F67
LINK.UTIL3.CNTL	F26	MERGERV1.FORT	F68
ROADMAP.CLIST	F27	NEXTV1.FORT	F69
SELECT.CLIST	F28	POLISHV1.FORT	F70
SIZER.CLIST	F29	PROMTV1.FORT	F71
UTIL.CLIST	F30	SELRAPV1.FORT	F72
UTIL.EQN.CLIST	F31	PSIZEDV2.FORT	F73
UTIL.REPORT.CLIST	F32	PSIZERV1.FORT	F74
UTIL.REPORT.TERMINAL	F33	SIZEMNV1.FORT	F75
EMPMAKV1.CLIST	F34	PUTIL1V1.FORT	F76
SELECT.RETURN.CLIST	F35	PUTIL2V2.FORT	F77
SELECT.REPORT.CLIST	F36	PUTIL3V2.FORT	F78
DUT132V2.FORT	F37	PUTIL5V1.FORT	F79
DUT216V1.FORT	F38	ROADMPV1.FORT	F80
DUT332V2.FORT	F39	RPT1V3.FORT	F81
SYMBOLV1.FORT	F40	RPT2V4.FORT	F82
DLNKZ1V1.FORT	F41	SELCOMV1.FORT	F83
LINK.LINKER.CNTL	F42	SELECTV1.FORT	F84
		EMPMAKV1.FORT	F85

6.0 DESCRIPTION OF MSFC SIGMA 9 COMPUTER SYSTEM

The MSFC SIGMA 9 computer is located on the first floor, A wing east, of Building 4487. It is a relatively old computer when compared with the current technology now used to build computers. However, it has been maintained and upgraded by MSFC and contains many features of the current technology. MSFC employs a full-time staff of government and contractor personnel to provide hardware and software maintenance. The SIGMA 9 is used by MSFC to solve many types of computational problems. These problems range from engineering calculations and design drawing updates to clerical text editing. The SIGMA is able to meet these requirements and it is a key element in MSFC's computer systems capability.

6.1 SIGMA 9 HARDWARE AND SOFTWARE

The SIGMA 9 computer permits both batch processing and multi-user time-sharing. The only batch operations employed in the MPPM installation work was the initial work required to deliver the IR&T source tape to the SIGMA operations staff and the periodic retrieval of printed outputs. All other development work was performed in the time-sharing mode. In order to support the time-sharing operation, the SIGMA has a main memory of 512K bytes and a large amount of secondary storage, both on disk and magnetic tape. Both forms of secondary storage were utilized during this effort. The SIGMA is set up to function as a virtual memory computer. This means that a user can use any amount of computer memory and the operating system will automatically use secondary storage to accommodate the program. The largest amount of memory a user could have at one time would be on the order of 332K bytes in any single "page" of memory. There are over 500,000,000 bytes of online disk storage available and the amount available to a single user depends upon the amount allocated by the operating staff.

Three types of terminals were used for interactive processing. These devices were a Tektronix 1014-1 terminal, a LSI terminal, and a Diablo teletypewriter terminal. All of these terminals were connected to the SIGMA over either a dedicated link or a telephone connection. These terminals allowed the user to access the SIGMA and perform any required processing as though the user were the only one working on the computer. The Tektronix terminal had a hard copy

output unit which allow the terminal display screen to be copied on paper in a manner similar to an office copying machine.

The SIGMA software capabilities represent the full line of requirements in order to utilize the hardware easily and efficiently. The SIGMA operating system, called Control Program-Five (CP-V), provides for five concurrent modes of operations:

- time-sharing,
- batch processing,
- remote processing,
- real-time processing,
- transaction processing.

In the time-sharing mode, CP-V allows up to 128 on-line terminals. These on-line terminals may use any of a number of on-line processors. The processors used in the MPPM installation include:

TEL - Terminal Executive Language - executive language control of all terminals activities.

EDIT - File Edit Program - composition and modification of programs and other bodies of text.

FORT4 - Xerox Extended FORTRAN IV - compilation of programs.

FDP - FORTRAN Debug Program - debugging of extended FORTRAN IV programs.

PCL - Peripheral Conversion Language - transfer and conversion of data between peripheral devices.

LINK - Program Linker - linkage of programs for execution.

All of these processors were used to install MPPM on the SIGMA computer.

6.2 IBM 370 DEPENDENT MPPM FEATURES

The IBM 370 computer is a large scale scientific/commercial computer system with many advanced hardware features. It utilizes both batch and time-sharing operation modes to permit efficient usage of its capabilities. The SIGMA 9 is a relatively "old" computer when compared with the IBM 370 but, as previously indicated, it has been maintained and upgraded by MSFC for engineering applications.

The MPPM programs were designed to take advantage of the IBM 370 hardware and software capabilities. Some of these features were unavailable on the SIGMA.

The main difference between the IBM 370 and the SIGMA is the byte addressable capability of the 370 as compared with the word addressable capability of the SIGMA. Each alphanumeric character requires one byte of memory on both the 370 and the SIGMA. With the 370, each character can be referenced or addressed in the high order language FORTRAN. However, on the SIGMA, only a word can be addressed. This means that a group of four characters must be accessed at a single memory reference, and that any character manipulation routines must be rewritten to reflect the SIGMA capabilities.

Each byte (character) in the memory of an IBM 370 can be addressed by having the FORTRAN compiler use logical variables which may be specified to require byte in memory. This specification can be defined through the use of a FORTRAN type statement: LOGICAL*1 VARNAME. The FORTRAN program variable VARNAM would then require only a single byte of memory. Alphanumeric (character) values may then be manipulated directly through VARNAM. Data values may be set via a read statement, either formatted or unformatted. or by a FORTRAN assignment statement, i.e., VAR1=VAR2. As long as the receiver of the assignment, VAR1, and the expression on the right side of the assignment symbol, the "=", are of the same FORTRAN type, in this case, type LOGICAL. The SIGMA 9 also permits the use of LOGICAL variables, but since it is a "word addressable" rather than an "byte addressable" machine, it does not permit the LOGICAL*1 form.

Additional differences between the FORTRAN compilers were also identified. The IBM 370 FORTRAN allows the use of list directed free format input where data items are required to be separated by a comma ",". The SIGMA also has this free format list directed input capability but the compiler implementations to accept alphanumeric data are different. The 370 automatically breaks a character string into the appropriate number of characters that represent the contents of a single word. A single variable may be specified to contain, 1, 4, or 8 characters by using the appropriate variable type. The SIGMA, requires that alphanumeric strings be enclosed within quotes ('xxx'), that a string be four characters or less, and that individual data items be separated by a comma ",". This problem with list directed input can be solved by inserting a comma "," between numeric items. However, the alphanumeric character strings must be converted to a fixed format input. This requires that the data files be restructured and that the FORTRAN READ statements be rewritten. Figure 6-1 shows an example of an IBM

```
03,23,  MATERIALS LIST(MAP LEVEL)
CONVENTIONS
  1.  INPUT CLASS CODES ARE:
      0=NEVER OCCURS AS INPUT
      1=MAY BE A PRIMARY OR SECONDARY INPUT
      2=OCCURS ONLY AS ASECONDARY INPUT
      4=MUST BE AN INTERMEDIATE MATERIAL
      -1=MAY BE A RAW MATERIAL OR AN INTERMEDIATE MATERIAL
  2.  OUTPUT CLASS CODES ARE:
      0=NEVER OCCURS AS OUTPUT
      1=MAY BE A PRIMARY OR SECONDARY OUTPUT
      2=OCCURS ONLY AS A SECONDARY OUTPUT
      3=MAY BE PRIMARY OR SECONDARY OUTPUT BUT NEVER PRIMARY OUTPUT OF EUP
      4=MUST BE AN INTERMEDIATE MATERIAL
      -1=MAY BE A FINAL PRODUCT OR AN INTERMEDIATE MATERIAL
  3.  DATA FORMAT
      COL  1 IS MATERIAL NODE NUMBER
      COL  2 IS INPUT CLASS CODE
      COL  3 IS OUTPUT CLASS CODE
      COL  4 IS NUMBER (N) OF MODULES TO WHICH MATERIAL IS INPUT
      COL  5 IS NUMBER (M) OF MODULES FROM WHICH MATERIAL IS OUTPUT
      NEXT N ENTRIES ARE THE INPUT MODULES
      NEXT M ENTRIES ARE THE OUTPUT MODULES
      MATERIAL NODE NAME FOLLOWS / (32 CHARACTERS MAX)
 1  1  1  2  1  22,33,  1/RAW LOW-BTU GAS
 2  1  1  1  5  14,  2,3,4,42,26/RAW MED-BTU GAS
 3 -1  0 14  0  1,3,10,19,22,26,44,37,36,34,54,24,25,30,  0/AIR
 4 -2  0  2  0  36,40,  /RAW WATER
 5  2  1 13  1  1,2,3,22,26,45,24,25,30,36,42,14,43,  60/PROCESS WATER
 6 -1  0  2  0  19,31,  0/RUN-OF-MINE COAL,COKE
 7  1 -1  1  1  24,  5/CLEAN LIQUID BOILER FUEL
 8 -1  0  3  0  17,38,31,  0/MINERALS OR CHEMICALS
 9  0 -1  0  1       9/SUBSTITUTE NATURAL GAS
10 -2 -1  2  1  14,43,  10/METHANOL
11  1  2  1  2  34,  12,15/PURGE GAS
12  0 -1  0  1    12/AMMONIA
13  1  1  1  1  43,  13/SYNTHESIS GAS
14  1 -1  1  3  13,  10,14,39/SYNTHESIS GAS
15  1 -1  1  1  12,  15/HYDROGEN
16 -1 -1  8  1  1,2,3,4,5,20,23,40,  16/PREPARED COAL
17  2  1  5  1  2,3,31,24,23,  17/PREPARED MINERALS
18  2  1  3  1  2,26,42,  18/OXYGEN
19 -1 -1  1  1  16,  19/RAW COAL
20  1  2  1  1  2  16/COAL FINES
21 -1  0  1  0  24, 0/NATURAL GAS
22  1 -1  2  1  37,38,  22/HOT LOW-BTU GAS
23  1  2  1  1  32,      31/302-HIGH GAS
24  1  1  1 11  31,  16,22,30,44,37,3,39,30,24,23,40/FLUE GAS
25  1  1  1  2  35,  24,30/HIGH TEMP THERMAL ENERGY
26 -1  0  1  0  26    /UNDERGROUND COAL
```

FIGURE 6-1 IBM DATA FILE

data file and Figure 6-2 shows the same data file after restructuring for the SIGMA. Each record of the IBM 370 file became three records on the SIGMA. The first record contains a number which specifies the number of data items to be read with a free-format READ; the second record contains the free-format data to be read; and the third record contains the alphanumeric data to be read under a formatted read. Each data file was restructured according to its own requirements. This example shown represents the general approach used.

These were the two major differences between the IBM 370 and the SIGMA 9 and caused the majority of the installation problems.

```
03,23,  MATERIALS LIST(MAP LEVEL)
CONVENTIONS
  1.  INPUT CLASS CODES ARE:
      0=NEVER OCCURS AS INPUT
      1=MAY BE A PRIMARY OR SECONDARY INPUT
      2=OCCURS ONLY AS ASECONDARY INPUT
      4=MUST BE AN INTERMEDIATE MATERIAL
      -=MAY BE A RAW MATERIAL OR AN INTERMEDIATE MATERIAL
  2.  OUTPUT CLASS CODES ARE:
      0=NEVER OCCURS AS OUTPUT
      1=MAY BE A PRIMARY OR SECONDARY OUTPUT
      2=OCCURS ONLY AS A SECONDARY OUTPUT
      3=MAY BE PRIMARY OR SECONDARY OUTPUT BUT NEVER PRIMARY OUTPUT OF EUP
      4=MUST BE AN INTERMEDIATE MATERIAL
      -=MAY BE A FINAL PRODUCT OR AN INTERMEDIATE MATERIAL
  3.  DATA FORMAT
      COL  1 IS MATERIAL NODE NUMBER
      COL  2 IS INPUT CLASS CODE
      COL  3 IS OUTPUT CLASS CODE
      COL  4 IS NUMBER (N) OF MODULES TO WHICH MATERIAL IS INPUT
      COL  5 IS NUMBER (M) OF MODULES FROM WHICH MATERIAL IS OUTPUT
      NEXT N ENTRIES ARE THE INPUT MODULES
      NEXT M ENTRIES ARE THE OUTPUT MODULES
      MATERIAL NODE NAME FOLLOWS / (32 CHARACTERS MAX)
 8
 1, 1, 1, 2, 1, 22,33, 1
/RAW LOW-BTU GAS
 11
 2, 1, 1, 1, 5, 19,  2,3,4,42,20
/RAW MED-BTU GAS
 19
 3,-1, 0,14, 0, 1,3,16,18,22,26,44,37,30,34,39,24,25,30, 0
/AIR
 7
 4,-2, 0, 2, 0, 36,   80
/RAW WATER
 19
 5, 2, 1,13, 1, 1,2,3,22,26,45,24,25,30,36,42,14,43, 80
/PROCESS WATER
 7
 6,-1, 0, 2, 0, 19,31,  0
/RUN-OF-MINE COAL,COKE
 7
 7, 1,-1, 1, 1, 24,  5
/CLEAN LIQUID BOILER FUEL
 8
 8,-1, 0, 3, 0, 17,34,31,   0
/MINERALS OR CHEMICALS
 6
 9, 0,-1, 0, 1,   9
```

FIGURE 6-2 SIGMA DATA FILE

7.0 INSTALLATION PROCEDURE ON SIGMA 9

The following is a description of the procedures used to install MPPM on the MSFC SIGMA 9 computer. This work was initiated prior to the delivery of the source code tape from IR&T.

A study of the SIGMA software and operations procedures was made to insure efficient usage of the SIGMA. This study included familiarization with the SIGMA FORTRAN compiler, the text editor, and the operating system. A list of the referenced SIGMA documentation is shown in Table 7-1.

TABLE 7-1

SIGMA DOCUMENTATION

XEROX Extended FORTRAN IV
- Operations Reference Manual
XEROX TEXT
- Language and Operations Reference Manual
XEROX FORTRAN Debug Package (FDP)
- Reference Manual
XEROX Control Program-Five (CP-V)
- Time-Sharing Reference Manual
- Time-Sharing User's Guide
- Transaction Processing Reference Manual

The Xerox Extended FORTRAN IV language processor (FORT4) consists of a comprehensive algebraic programming language, a compiler, and a large library of subroutines. The language is a superset of most available FORTRAN languages, containing many extended language features to facilitate program development and checkout. The compiler is designed to produce very efficient object code, thus reducing execution time and core requirements, and to generate extensive diagnostics to reduce debugging time. The library contains over 180 subprograms and is available in a reentrant version. Both the compiler and runtime library for object programs are reentrant programs that are shared among all concurrent users to improve the utilization of the critical core resources.

The principal features of Xerox Extended FORTRAN IV are as follows:

- Extended language features to reduce programming effort and increase range of applications.
- Extensive meaningful diagnostics to minimize debugging time.

- In-line assembly language code to reduce execution time of critical parts of the program.
- Overlay organization for minimal core memory utilization.
- Compiler produced reentrant programs.
- Full use of CP-V features.
- Availability of reentrant version of library.

The study of the SIGMA FORTRAN compiler indicated several major problem areas when compared with the IBM 370 FORTRAN G compiler which was used to implement MPPM. As mentioned previously, the two main problem areas were in the use of LOGICAL variables to manipulate character string data and the use of the list directed free format input.

The Edit processor is a line-at-a-time context editor designed for on-line creation, modification, and handling of programs and other bodies of information. All Edit data is stored on RAD or disk pack storage in keyed file structure of sequence-numbered variable length records. This structure permits Edit to directly access each line or record of data.

Edit functions are controlled through single-line commands supplied by the user. The command language provides for insertion, deletion, reordering, and replacement of lines or groups of lines of text. It also provides for selective printing, renumbering records, and context editing operations of matching, moving, and substituting line-by-line within a specified range of text lines. File maintenance commands are also provided to allow the user to build, copy, and delete whole files of text lines.

The SIGMA text editor capabilities were adequate for the installation task. The main editor capabilities required were to locate a character string and to replace that character string with another. The text editor was an interactive program which allowed the editing work to be performed at an interactive terminal in a time-sharing mode of operation.

The SIGMA operating system, called Control Program V, CP-V, was a relatively old form of operating system when compared with operating systems available today, but it too proved to be adequate for the task. CP-V was able to support a multi-user, multi-task mode of operating which permitted the use of time-sharing. The tape and disk file manipulation features of CP-V were also extremely useful and relatively easy to learn and use.

7.1 SOURCE TAPE CHARACTERISTICS

The source code tape which was furnished by IR&T contained 85 data sets which included three main "types" of data. These data sets were named according to the following conventions:

NAME.TYPE

where NAME indicated the MPPM module and function to which the data set belonged and TYPE referred to the type of the data set. Data set names and types are shown in Table 5-2. A TYPE of DATA indicated a MPPM data file. An example of a DATA file was shown in Figure 6-1. The original MPPM data file formats may be found in the MPPM User's Guide, Chapter 10. A TYPE of CLIST indicated an IBM 370 Job Control Language (JCL) stream used to run a particular MPPM module. These JCL streams were included to define the file movement between programs. An example of a CLIST data set is given in Figure 7-1. Finally, a TYPE of FORT indicated a FORTRAN code file. Data sets number 2 through 8 contain MPPM data files, data set 1 and data sets 9 through 36 contain JCL command streams; and data sets 37 through 85 contain MPPM FORTRAN source code. There are several miscellaneous data set types that were included. Data set COSTER.REPORT. TERMINAL, for example, contains the JCL to produce the COSTER module report on the user terminal. Data set LINK.SIZER.CNTL contains a FORTRAN program which essentially controls the functioning of the SIZER program in the MPPM LINKER module.

A copy of the tape received from IR&T was made (called TAPE 1) and is available from SRS. This tape is reel number FT#MPPM. Every data set was blocked at 3120 characters. The CLIST data sets were variable blocked with logical record lengths of 255 characters, and all other data sets were fixed blocked with a logical record length of 80 characters. This tape format was used to insure the MPPM files would fit on a single magnetic tape. The use of variable block sizes for the CLIST's was an option of IR&T and caused a minor problem in the installation. The use of the fixed block size corresponded to a card image per logical record of 80 characters. The use of 255 characters per logical record allowed the CLIST's to be efficiently written on the tape but caused a problem on the SIGMA which could not decode the blocking information in its general purpose utility routines. A special program had to be written to decode and deblock the CLIST data sets. It should be noted that this problem would not arise if MPPM were to be installed on another IBM computer.

```
PROC 0
WRITE
WRITE BEGIN ROADMA  &SYSDATE, AT &SYSTIME
CONTROL NOMSG NOFLUSH
FREE DA('CN6449.MM9.MPPM.ALTMAT.RMAP.DATA')
FREE F(FT10F001 FT08F001 FT16F001)
FREE F(FT12F001 FT13F001 FT20F001 FT11F001)
FREE F(FT05F001 FT06F001)
FREE ATTRLIST(A B C)
ATTRIB A RECFM(F B) LRECL(80) BLKSIZE(3120)
ATTRIB B INPUT
ATTR C RECFM(F) LRECL(80) BLKSIZE(80)
DEL ROADMAP.DATA
DEL MATNAM.DATA
DEL ROADMAP.TRACE.DATA
CONTROL MSG NOFLUSH
ALLOC F(FT08F001) DA(ROADMAP.TRACE.DATA) SPACE(10,5) TRACKS -
 UNIT(3350) USING(A) RELEASE
ALLOC DA(*) F(FT05F001) USING(C)
ALLOC DA(*) F(FT06F001)
ALLOC DA('NULLFILE') F(FT10F001)
ALLOC DA('CN6449.MM9.MPPM.ALTMAT.RMAP.DATA') F(FT11F001) SHR USING(B)
ALLOC DA('CN6449.MM9.MPPM.FUNMOD.RMAP.DATA') F(FT12F001) SHR USING(B)
ALLOC DA('CN6449.MM9.MPPM.MATNODE.RMAP.DATA') F(FT13F001) SHR USING(B)
ALLOC DA(ROADMAP.DATA) NEW F(FT20F001) SPACE(133,1) -
     BLOCK(1680) UNIT(3350)
ALLOC DA(MATNAM.DATA) F(FT16F001) NEW UNIT(3350) USING(A)
CALL 'CN6449.MM9.MPPM.LOAD(ROADMAP)'
WRITE
WRITE ROADMAP ROUTES HAVE BEEN WRITTEN TO FILE: ROADMAP.TRACE.DATA
WRITE THIS FILE MAY BE EDITED NONUM VIA TSO OR SUPERWYLBUR.
WRITE
CONTROL NOMSG NOFLUSH
FREE DA(ROADMAP.DATA MATNAM.DATA)
FREEALL
EXIT
```

FIGURE 7-1 SAMPLE CLIST DATA SET

ORIGINAL PAGE IS
OF POOR QUALITY

7.2 DATA SET MANIPULATIONS

The original data tape (TAPE 1) was then read to build data files acceptable to the SIGMA. These files were saved on disk. FORTRAN code files were then combined as shown in Table 7-2 to build the various MPPM program modules. These program modules represent individual MPPM processing functions which the user may request to execute. The structure of these program modules is shown in Figure 7-2. These modules were saved on tape (TAPE 2). This tape is reel number LT#M1. It should be noted that this tape contains data sets that would be usable on an IBM computer.

7.3 MPPM MODIFICATIONS REQUIRED

Each type of data set, i.e., CLIST, DATA, and FORT required a different installation effort since each contained different types of data. The work performed on each type of data set is described below.

7.3.1 CLIST DATA SETS

As described previously, the data sets of type CLIST contain IBM 370 JCL command streams which define the file manipulation and access sequence and modes required to run a particular MPPM module. These JCL streams are called PROCEDURES or PROCS and contain JCL streams that are used frequently. The CLIST shown in Figure 7-1 contains the JCL required to run the ROADMAP MPPM module. This CLIST will be used as an example to demonstrate the conversion work required for all CLIST's.

An equivalent command stream must be generated on the SIGMA to implement the same overall operating system functions. The SIGMA CP-V command stream is called a data control block (DCB). First, the 370 procedure must be analyzed as to the functions it performs. These functions are essentially (1) to FREE or release a group of data sets, (2) to define the physical structure of several new data sets, (3) to delete previously generated output data sets, (4) to attack and allocate data files required to run ROADMAP, (5) to load and run the ROADMAP program, and (6) to free or release all data sets when execution terminates. Corresponding CP-V commands are built and stored in a DCB to run ROADMAP. The SIGMA 9 DCB for the example CLIST is shown in Figure 7-3.

TABLE 7-2

MPPM PROGRAM MODULE CONSTRUCTION

MPPM PROGRAM MODULE	DATA SET NAME(.FORT)	MPPM PROGRAM MODULE	DATA SET NAME(.FORT)
ROADMAP	ROADMPV1	UTIL3	UTIL3MV3
			PUTIL3V2
MAKMAT	MAKMATV1		PUTIL5V1
			JQUERYV1
MATEDT	MATEDTV1		PSIZEDV2
			EQDOXV1
SELECT	SELECTV1		EQDOZV1
MAKFIG	CONFIGV1	COSTER	COSTCOV3
	LQUERYV1		JQUERYV1
	SYMBOLV1		EQDOXV1
	NEXTV1		
	POLISHV1	UTILRPT	RPT1V3
SELCOM	SELCOMV1	COSTRPT	RPT2V4
LINKER	LINKV1	MLNKDMP	DLNK21V1
	LINK2V1		
	EQDOV1	SLNKDMP	DLNK31V1
	PROMTV1		
	JQUERYV1	SIZDMP	DSIZ29V1
MERGER	MERGER	UIDMP	DUT132V2
DMPEUP	DMPEUPV1	UZDMP	DUT216V1
SIZER	SIZEMNV1	U3DMP	DUT332V2
	SIZELV1		
	PSIZERV1	SELREP	SELREPV1
	JQUERYV1		
	EQDOZV1		
UTILI	VANDSDV3		
	MULILV2		
	JQUERYV1		
	EQDOXV1		
	WATERV2		
UTIL2	UTIL2MV2		
	PUTIL1V1		
	PUTIL2V2		
	EQDOV1		
	PROMTV1		
	JQUERYV1		

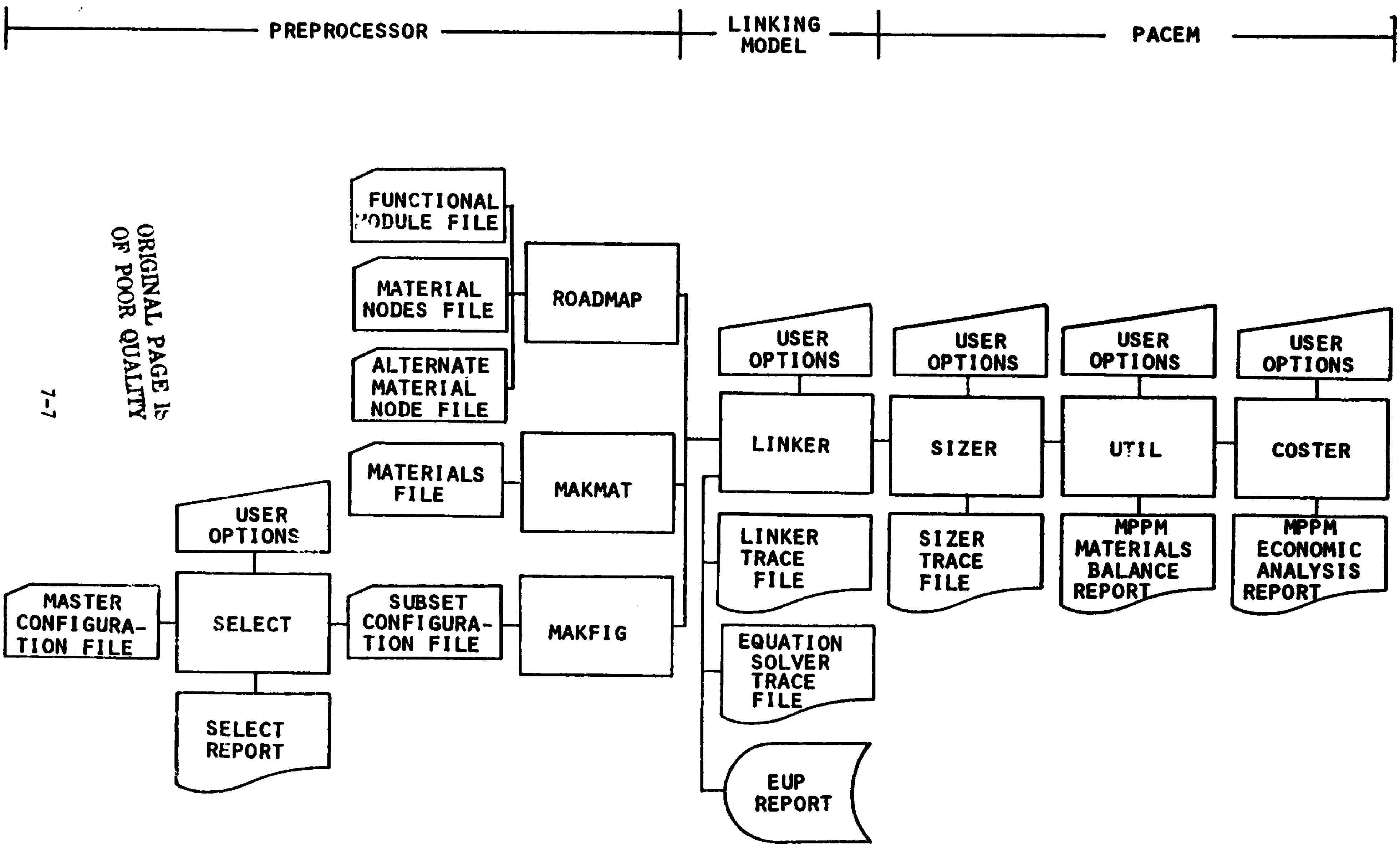

FIGURE 7-2 MPPM PROGRAM STRUCTURE

ORIGINAL PAGE IS
OF POOR QUALITY

```
SETDCB
XTY
     .500 !ECHO
    1.000 !R
    2.000 !LIST ON ME
    3.000 !SET F:6/FILE6;OUT;SAVE
    3.100 !SET F:5 ME
    4.000 !SET F:11/ F4;IN
    5.000 !SET F:12/ F2;IN
    6.000 !SET F:13/FIXF3;IN
    6.100 !SET F:16/TEMP1;OUT;SAVE
    7.000 !SET F:10/MNODNM;OUTIN;SAVE
    8.000 !SET F:8/FCHAIN;OUTIN;SAVE
    9.000 !SET F:20/ROUTEM;OUTIN;KEYED;KEYM=3;DIRECT;SAVE
    9.200 !LYNX B:ROADMP OVER L:ROADMP
   11.000 !L:ROADMP.
X
```

FIGURE 7-3 SIGMA DCB EXAMPLE

7.3.2 DATA DATA SETS

The problems with the DATA data sets have been outlined previously. As each DATA data set was encountered in getting an MPPM module to run on the SIGMA, an analysis was made to determine the most efficient manner in which to be able to READ that data file. As shown in Section 6.2, the options were (1) to reformat the data file, or (2) to alter the data file READ statement. Either or both of these options were employed to overcome the limitations of the SIGMA FORTRAN format-free list directed input. Table 7-3 shows the status of the data files.

TABLE 7-3

DATA FILE STATUS

INPUT DATA FILE	STATUS
FUNMOD.RMAP.DATA(F2)	Converted
MATNODE.RMAP.DATA(F3)	Converted
ALTMAT.RMAP.DATA(F4)	Converted but contains apparent error
ASPTAB.DATA(F5)	Unconverted
NEW.MASTER.DATA(F6)	Converted
MAKMAT.MIX.DATA(F7)	Converted
MAKMAT.SUB.DATA(F8)	Converted

7.3.3 FORTRAN DATA SETS

The conversion of the MPPM FORTRAN source code was undertaken in the following manner. Each of the 23 program modules listed in Table 7-2 was compiled and a listing along with any compilation errors was obtained. As expected, this revealed that extensive use of the IBM 370 FORTRAN G language extensions and hardware dependent features had been made. The first step was to remove all FORTRAN compilation errors from all the source files. This mainly involved the conversion of all LOGICAL*1 type statements to LOGICAL type statements. Several other minor compilation errors were encountered but these appeared to be simple oversights rather than critical system design problems.

Next a study of the general MPPM structure as shown in Figure 7-2 indicated that the most efficient manner in which to bring MPPM on-line would

be to get the following modules running in the order shown in Figure 7-4. The strategy behind this schedule was that the experience gained in the early conversion work, i.e., ROADMAP, MAKMAT, and SELECT which could be manipulated in parallel efforts, would be extremely useful when the larger and apparently more difficult modules, i.e., LINKER and COSTER, were encountered. Table 7-4 shows the status of the various program modules at the termination of the installation effort and Table 7-5 shows the approximate size of each module.

Most of the effort during the conversion process took place in the area of the character manipulation functions. The change from one character per word to the four characters per word of the SIGMA proved to be very time consuming. Many of the problems produced were not immediately obvious and sometimes required several days to trace an erroneous output back to its original source. In several cases, data constants which were character strings had to be modified to reflect different implementations of the manner in which character strings are initialized on the IBM 370 and on the SIGMA 9.

MPPM may be logically divided into two main components. The first component is in the area of system definition. This corresponds to the modules of ROADMAP, SELECT, MAKFIG, and MAKEDT. These modules are for the most part nothing more than character manipulation programs. They do very little numerical processing. The second component performs the model evaluation and is comprised of LINKER, SIZER, UTIL, and COSTER. Almost no time was spent in converting these routines since they cannot run without the output of the first component. It was expected that the installation of the second component of MPPM would be essentially error-free.

At the termination of this installation effort, all changes to the MPPM files were saved on a magnetic tape. This tape (TAPE 3) has the name LT#M2 and is available from SRS. The overall manipulations of the MPPM files is shown in Figure 7-5.

It should be noted again that TAPE 1 contains a copy of the original files as provided by IR&T; TAPE 2 contains a restructured form of the original tape in which various files have been combined into logically related MPPM system components; and that TAPE 3 contains the data files as they appeared at the

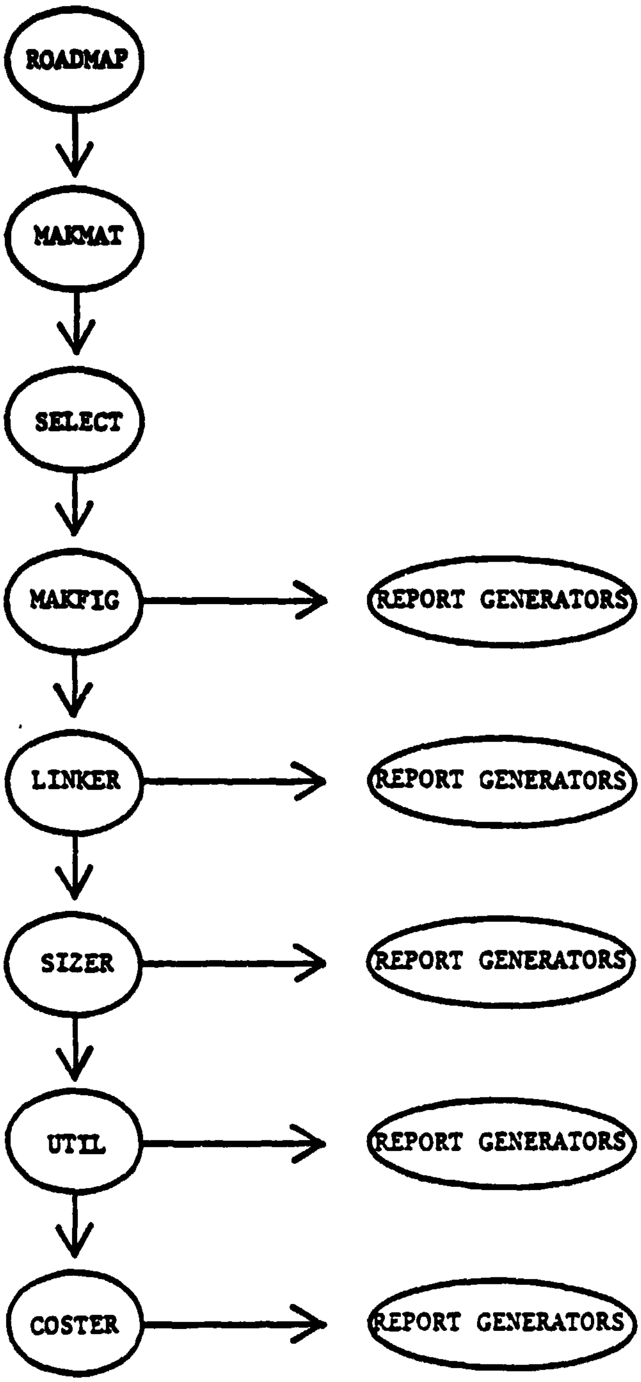

FIGURE 7-4 MFPM IMPLEMENTATION DIAGRAM

TABLE 7-4

PROGRAM MODULE STATUS

MODULE	STATUS
ROADMAP	Runs but encounters an apparent data file error.
MAKMAT	Partially converted.
MATEDT	Compiles
SELECT	Runs successfully
MAKFIG	Partially converted
SELCOM	Compiles
LINKER	Compiles
MERGER	Compiles
DMPEUP	Compiles
SIZER	Compiles
UTIL1	Compiles
UTIL2	Compiles
UTIL3	Compiles
COSTER	Compiles
UTILRPT	Compiles
COSTRPT	Compiles
MLNKDMP	Compiles
SLNKDMP	Compiles
SIZDMP	Compiles
U1DMP	Compiles
U2DMP	Compiles
U3DMP	Compiles
SELREP	Compiles

TABLE 7-5

MPPM MODULE SIZE

MPPM MODULE	STATEMENTS
ROADMAP	717
MAKMAT	1,760
MATEDT	412
SELECT	1,065
MAKFIG	4,181
SELCOM	330
LINKER	8,658
MERGER	741
DMPEUP	117
SIZER	5,460
UTIL1	7,332
UTIL2	9,048
UTIL3	7,098
COSTER	4,836
UTILRPT	1,014
COSTRPT	1,755
MLNKDMP	273
SLNKDMP	273
SIZDMP	273
U1DMP	273
U2DMP	312
U3DMP	312
SELREP	117
TOTAL	56,357

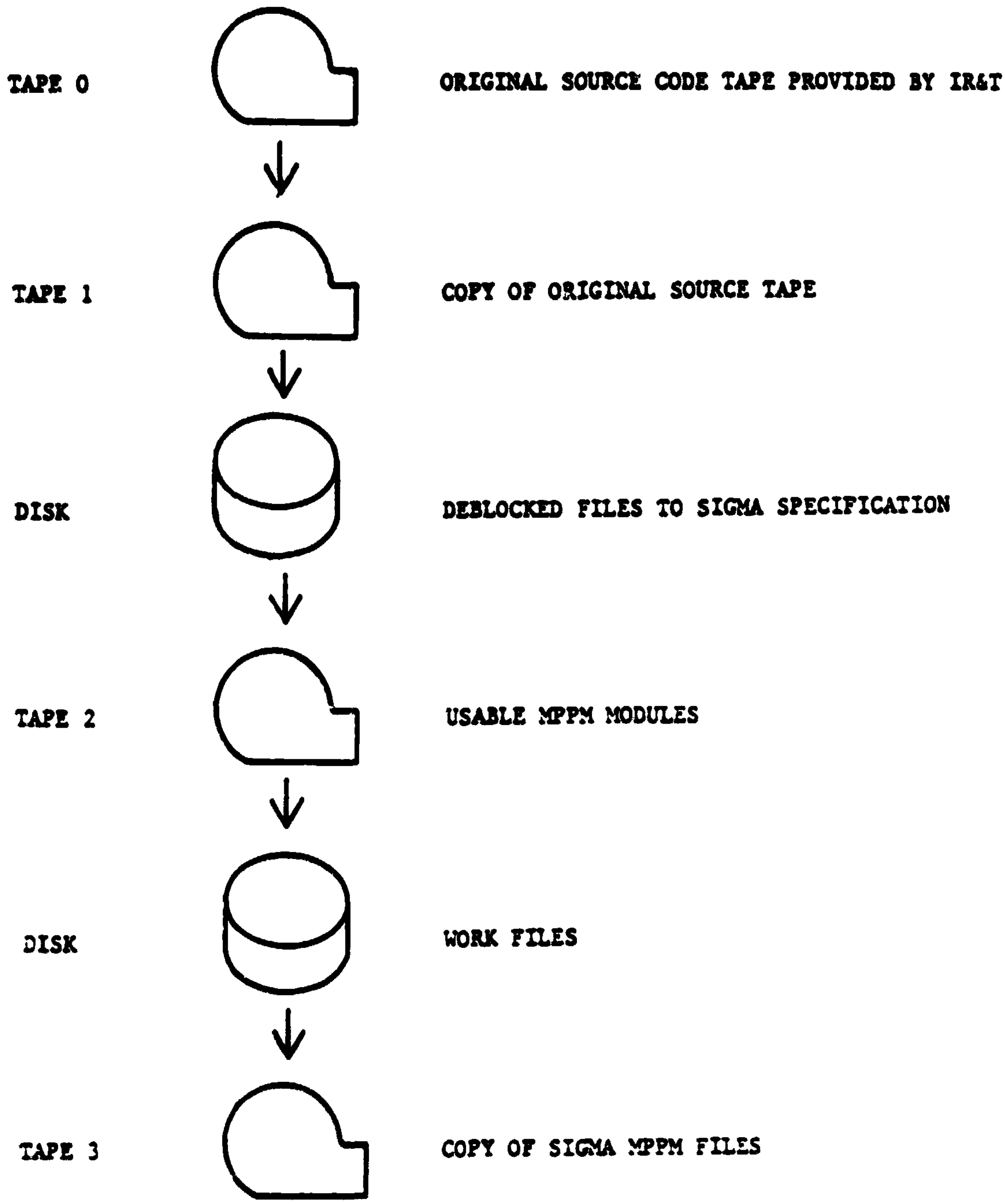

FIGURE 7-5 MPPM SIGMA TAPE FLOW

termination of this work. TAPE 2 is in the format required for installation on any other IBM computer, while TAPE 3 is in the format required for further SIGMA installation work.

8.0 SUMMARY

The MPPM program has progressed through a multi-year three-phase development effort and has been extensively tested. It provides a top-level evaluation tool and, with module enhancements as described in Volume I, can be an effective coal conversion systems design model.

The work performed to install the MPPM program on the MSFC SIGMA 9 computer has been described in this Appendix. The installation was only partially completed due to the reduction in scope of this project activity. However, a major portion of the effort was accomplished. The entire program was compiled on the SIGMA after certain modifications were implemented. Details of the installation of the MPPM, CLIST, DATA, and FORTRAN type data sets have been described to illustrate the actual work performed. The installation of the remaining portions of MPPM on the SIGMA would involve modifications similar to those made to the modules previously installed. The second part of MPPM, model evaluation calculations, should be relatively straight forward to install. No computational type errors were found in the modules installed.

A modification to the MPPM code is recommended. The subroutine FLD, which performs most of the character manipulation functions, should be rewritten. There are several versions of this subroutine in the MPPM code. All of these versions should be consolidated into one FLD subroutine.

The installation of MPPM on the TVA's IBM computer system could proceed using the work performed on the SIGMA installation as a basis. The experience gained during the SIGMA work would be extremely useful for an IBM installation effort. The original code received from IR&T, Tape 1 or 2, would be appropriate for an IBM application since the modifications made to the program for the SIGMA would not be required.

The MPPM program was installed on an IBM 370 at Argonne National Laboratory (ANL). Discussions with ANL personnel indicated that few problems were encountered during this installation. Generally, the code and data files were error free and the only changes required were modifications to the JCL structures to reflect slightly different operating systems.